An Improved Nearest Neighbor Based Entropy Estimator with Local Ellipsoid Correction and its Application to Evaluation of MCMC Posterior Samples

Chien Lu

University of Tampere
Faculty of Natural Sciences
M.Sc. Thesis
Supervisor: Jaakko Peltonen

University of Tampere
Faculty of Natural Sciences
Master's Degree Programme in Computational Big Data Analytics
Machine Learning Applications in Mortgage Defaults Predictions
CHIEN LU: An Improved Nearest Neighbor Based Entropy Estimator with Local Ellipsoid Correction and its Application to Evaluation of MCMC Posterior Samples
M.Sc. thesis, 37 pages
July 2018

Abstract

Entropy estimation is an important technique to summarize the uncertainty of a distribution underlying a set of samples. It ties to important research problems in fields such as statistics, machine learning and so on. The k-nearest neighbor (kNN) estimator is one widely used classical nonparametric method although it suffers bias issue especially when the dimensionality of the data is high.

In this thesis, an improved kNN entropy estimator is developed. The proposed method has the advantage of a learning a local ellipsoid to be used in the estimation, in order to mitigate the bias issue which results from the local uniformity. Several numerical experiments have been conducted and the results have shown that the proposed approach can efficiently reduce the bias especially in when the dimension is high.

Another studied topic in this thesis is the evaluation of the correctness of the posterior samples when conducting Bayesian inferences. This thesis demonstrates that the proposed estimator can be applied to such a task. We show that the simulation-based approach is more efficient and discriminative than a lower bound based method by one simple experiment, and the proposed kNN estimation can improve the accuracy of the state-of-the-art simulation-based approach.

Keywords: entropy estimation, nonparametric estimator, Bayesian inference

Contents

1. Introduction

Entropy has been one of the most important numerical quantities in statistics, machine learning and other disciplines such as Physics. It provides a summary measurement of the degree of uncertainty of a system, and the notion is also perceived as mean "information" provided by locations of individual samples. In theory, to obtain the value of Entropy of a system, the definition of the underlying probability distribution is required, that is, the probability density function (PDF) needs to be available. However, in most of the real-world cases, it is common that the underlying PDF is not always available, that is, only the samples are observed. This raises the research problem of estimating entropy without a clear definition of the underlying PDF of the observed data which is known as non-parametric entropy estimation.

The main challenge of non-parametric entropy estimation is how to estimate the underlying probability density of data points as accurately as possible with solely the observed data in hand. Many estimators such as k-nearest neighbor (kNN; Kozachenko, L. F., & Leonenko, 1987) estimator and kernel density estimator (KDE, Silverman, 1986) or hybrid methods such as the Orava's approach(Orava, 2011) have been proposed. This research focuses on the kNN and its relevant approaches.

For the classical kNN method, it has been shown that the method is able to deliver promising results in lower dimensional cases, whereas in higher dimensional cases the classical kNN estimator often yields biased results. One possible explanation of the phenomena is that it is due to the basic assumption of kNN, which considers the data points are almost uniformly distributed inside of the hypersphere around the interested data point. The hypersphere structure is not capable of capturing the twisted shape of the observed data especially in higher-dimensional cases.

There are several approaches aiming at solving the bias problem of the classical kNN estimator. In this thesis, the current developed works on this issue are discussed. A simulation-based comparative study is also provided. This research further provides a solution called the ellipsoidal corrected kNN (EC-kNN) estimator to ease the problem resulting from the above-mentioned assumption. The notion of ellipsoidal correction has been proposed by Gao et al.'s work (Gao et al., 2015), however, the procedure provided in Gao's work is not designed for kNN. More details of Gao's work will be discussed in other sections.

One of the applications of non-parametric entropy estimation is assessment of the correctness of posterior samples when conducting Bayesian inference. This research therefore discusses the basic notion of the usage of non-parametric entropy estimation in posterior samples evaluation. A demonstration of how to apply the proposed EC-kNN to the problem is provided.

The rest of this thesis is organized as follows. The background of the research including entropy, kNN estimator, related previous works and MCMC posterior samples evaluation are discussed in the next section. The notion of the proposed method and proposed algorithm are presented in Section 3. Section 4 provides the simulation-based experiments. Section 5 demonstrates the usage of the proposed estimator in a task of evaluating the correctness of MCMC posterior samples with a Bayesian linear regression example. The conclusions and discussion are given in Section 6.

2. Background

2.1 Entropy

Entropy is one of the most well-known approaches to quantify the uncertainty (or the amount of "information") of the data in hand. In this thesis, the main focus is on the Shannon's entropy (Shannon, 1948), named after Claude Shannon. It can be dated back to the middle of the twentieth century. It was first proposed in information theory and it has been applied in a variety of research areas such as such as statistics (e.g. Dudewicz et al., 1981; Joe, 1989), machine learning (e.g. Berger et al., 1996), finance (e.g. Gulko, 1999; Philippatos and Wilson, 1972), genetics (e.g. Fuhrman et al., 2000; Hampe et al., 2003) and so on. Mathematically, for a discrete random variable $\boldsymbol{X}$ generated from a probability distribution P, the Shannon's entropy is defined as

$$H(\boldsymbol{X}) = -\sum_{i} p(\boldsymbol{x_i}) \log p(\boldsymbol{x_i})$$

where H is named after Boltzmann's H-theorem and p denotes the probability mass function and where the $\boldsymbol{x_i}$ are the possible values of the random variable. On the other hand, while $\boldsymbol{X}$ is a continuous random variable with the probability distribution P, the differential entropy is then defined as

$$H(\boldsymbol{X}) = -\int p(\boldsymbol{x}) \log p(\boldsymbol{x})\, d\boldsymbol{x}$$

where p denotes the probability density function. This thesis focuses on the differential entropy, that is, the entropy of continuous variables.

Other than being a measure of uncertainty, the entropy is also related to other important measures in probability theory and information theory such as mutual information (see, e.g., Cover and Thomas, 2006) which measures the mutual dependence of two random variables and Kullback-Leibler divergence (KL divergence; Kullback and Leibler, 1951) which measures the dissimilarity between two random variables.

Mutual information measures the amount of mutual dependence of two random variables (the average information provided about one variable by knowing the value of the other variable) by

measuring the expected similarity of the joint probability distribution $p(\boldsymbol{x}, \boldsymbol{y})$ and the factored marginal distribution $p(\boldsymbol{x})p(\boldsymbol{y})$. The Mutual information of two random variables X and Y is defined as:

$$I(\boldsymbol{X}, \boldsymbol{Y}) = \int \int p(\boldsymbol{x}, \boldsymbol{y}) \log \frac{p(x,y)}{p(x)p(y)} d\boldsymbol{x}\, d\boldsymbol{y}, \tag{2.1.1}$$

furthermore, the equation (2.1.1) can be written as

$$\begin{aligned} I(\boldsymbol{X}, \boldsymbol{Y}) &= \int_Y \int_X p(\boldsymbol{x}, \boldsymbol{y}) \log \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} d\boldsymbol{x}\, d\boldsymbol{y} \\ &= \int_Y \int_X p(\boldsymbol{x}, \boldsymbol{y}) \log \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{y})} d\boldsymbol{x}\, d\boldsymbol{y} - \int_Y \int_X p(\boldsymbol{x}, \boldsymbol{y}) \log p(\boldsymbol{x})\, d\boldsymbol{x}\, d\boldsymbol{y} \\ &= \int_Y \int_X p(\boldsymbol{x}, \boldsymbol{y}) \log p(\boldsymbol{y}|\boldsymbol{x})\, d\boldsymbol{x}\, d\boldsymbol{y} - \int_X \log p(\boldsymbol{x}) \int_Y p(\boldsymbol{x}, \boldsymbol{y}) d\boldsymbol{y}\, d\boldsymbol{x} \\ &= \int_Y \int_X p(\boldsymbol{x}, \boldsymbol{y}) \log p(\boldsymbol{y}|\boldsymbol{x})\, d\boldsymbol{x}\, d\boldsymbol{y} - \int_X \log p(\boldsymbol{x}) \log p(\boldsymbol{x})\, d\boldsymbol{x} = -H(\boldsymbol{Y}|\boldsymbol{X}) + H(\boldsymbol{X}). \end{aligned}$$

Therefore, it's can be seen as the difference between the conditional entropy $H(Y|X)$ and the marginal entropy $H(X)$.

On the other hand, the KL divergence, which is also known as "relative entropy", has been one of the most well-known measure of the dissimilarity between two probability distributions. For two probability distributions P and Q with density functions p and q correspondingly, the KL divergence from Q to P is defined as:

$$KL(P||Q) = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} d\boldsymbol{x}. \tag{2.1.2}$$

Note that, equation (2.1.2) can be written as

$$\begin{aligned} KL(P||Q) &= -\int p(x) \log q(x)\, dx + \int p(x) \log p(x)\, dx \\ &= H(P, Q) - H(P) \end{aligned}$$

and the decomposition shows that the KL divergence can be seen as the difference between two entropy measures where $H(P, Q)$ is the cross entropy of P and Q and $H(P)$ is the entropy of P. KL divergence is usually taken as the measurement of the dissimilarity between two probability distributions, since it is always positive by definition, the zero occurs when the distributions P and Q are identical. Note that, the KL divergence is not always symmetric, which means $KL(P||Q)$ is not always equivalent to $KL(Q||P)$, in practice, another symmetrized divergence

$$KL(Q||P) + KL(P||Q)$$

based on KL-divergence is often used.

The KL divergence has been utilized for a variety of applications of Bayesian inference. In approximation-based inference approaches, it has been the foundation of the variational Bayes method (see Blei et al., 2017) and Expectation Propagataion (Minka, 2001). On the other hand, in sampling-based approaches, KL divergence has been used to evaluate the correctness of the posterior samples (Chauveau and Vandekerkhove, 2014; Cusumano-Towner and Mansinghka, 2016) in the sense of how well the set of samples represents the underlying posterior distribution. Chauveau et al. measure the KL divergence from the samples to the true posterior distribution whereas the work of Cusumano-Towner and Mansinghka takes advantage of the symmetric KL divergence.

2.2 K Nearest Neighbor Estimator

It's common that in some situations, only the generated data points $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots \boldsymbol{x}_i, \dots \boldsymbol{x}_N\}$ are observed but the underlying probability distribution P as well as the probability density function $p(\boldsymbol{x})$ are unknown. Under such a circumstance, several approaches have been proposed to obtain an estimation of the entropy value in nonparametric ways. Among them, one of the most well-known methods so far has been the k-nearest-neighbor (kNN) estimator, which was first proposed by Kozachenko and Leonenko (1987).

The classical kNN estimator begins with a Monte-Carlo estimator

$$H(\boldsymbol{X}) \approx -\frac{1}{N}\sum_{i=1}^{N}\log p(\boldsymbol{x}_i)$$

where the probability density p is unknown.

The kNN estimator then approximates the probability density $p(\boldsymbol{x}_i)$ by creating an ε-ball centered at $\boldsymbol{x}_i$, in a Euclidean space, which exactly contains k nearest neighbors of $\boldsymbol{x}_i$, thus the distance from $\boldsymbol{x}_i$ to its kth nearest neighbor is the radius ε. The volume of the opened ε-ball is

$$V_D = c_D\varepsilon^D$$

where $c_D = \frac{\pi^{\frac{D}{2}}}{\Gamma\left(1+\frac{D}{2}\right)}$, D is the dimension, Γ is the Gamma function and c_D is the volume of a D-dimensional unit ball ($\varepsilon = 1$).

The classical kNN estimator assumes the density to be uniform inside of the unit ball, which means the probability density function p inside of the small enough ε-ball is considered everywhere a constant. Mathematically, the above-mentioned notions can be written as,

$$p(\boldsymbol{x}_i) \times V_D \approx p_{\boldsymbol{x}_i}(\varepsilon) \tag{2.2.1}$$

Furthermore, consider the probability distribution $P_{\boldsymbol{x}_i}(\varepsilon)$ of ε where ε represents the distance from the $\boldsymbol{x}_i$ to its kth nearest neighbor. The value of $P_{\boldsymbol{x}_i}(\varepsilon)d\varepsilon$ is the probability that the kth nearest neighbor is right on the surface of the ball, more accurately, the value of $P_{\boldsymbol{x}_i}(\varepsilon)d\varepsilon$ is the probability of the condition that the distance from the $\boldsymbol{x}_i$to its kth nearest neighbor is between ε and $\varepsilon + d\varepsilon$; in the meanwhile, there are $k-1$ other points are falling inside of the ball and $N-k-1$ other points are falling outside of the ball. The above-mentioned condition can be depicted by a trinomial formula

$$P_{\boldsymbol{x}_i}(\varepsilon)d\varepsilon = \binom{N-1}{1}\binom{N-2}{k-1}\frac{dp_{\boldsymbol{x}_i}(\varepsilon)}{d\varepsilon}d\varepsilon\left(p_{\boldsymbol{x}_i}(\varepsilon)\right)^{k-1}\left(1-p_{\boldsymbol{x}_i}(\varepsilon)\right)^{N-k-1} \tag{2.2.2}$$

or

$$P_{\boldsymbol{x}_i}(\varepsilon) = k\binom{N-1}{k}\frac{dp_{\boldsymbol{x}_i}(\varepsilon)}{d\varepsilon}d\varepsilon\left(p_{\boldsymbol{x}_i}(\varepsilon)\right)^{k-1}\left(1-p_{\boldsymbol{x}_i}(\varepsilon)\right)^{N-k-1} \tag{2.2.3}$$

where $p_{\boldsymbol{x}_i}(\varepsilon)$ denotes the probability mass inside of the unit ball such that

$$p_{\boldsymbol{x}_i}(\varepsilon) = \int_{\mathcal{B}(\boldsymbol{x}_i,\varepsilon)} p(\boldsymbol{x})\, d\boldsymbol{x}$$

Then the expectation of the logarithm of the right-hand side of the equation (2.2.1), can be obtained by inserting the equation (2.2.3), thus,

$$\begin{aligned} E\left[\log p_{\boldsymbol{x}_i}(\varepsilon)\right] &= \int_0^\infty d\varepsilon\, P_{\boldsymbol{x}_i}(\varepsilon)\log p_{\boldsymbol{x}_i}(\varepsilon) \\ &= k\binom{N-1}{k}\int_0^\infty dp\, p^{k-1}(1-p)^{N-k-1}\log p = \psi(k) - \psi(N) \end{aligned} \tag{2.2.4}$$

where ψ is the digamma function, which is defined as:

$$\psi(x) = \log\left(\Gamma(x)\right) = \frac{\Gamma'(x)}{\Gamma(x)}.$$

Therefore, after taking the logarithm and computing the expectation on the both sides of the equation (2.2.1), it becomes

$$\log p(\boldsymbol{x}_i) + \log V_D \approx \psi(k) - \psi(N) \tag{2.2.5}$$

and after simple algebra, the classical kNN estimation of entropy can be obtained as:

$$H(\boldsymbol{X}) \approx \sum_{i=1}^{N} -\log p(\boldsymbol{x}_i) = \psi(N) - \psi(k) + \log(c_D) + \frac{D}{N}\sum_{i=1}^{N}\log \varepsilon_i.$$

The classical estimator has been widely applied in many different research problems. Since the entropy is related to other probability measurements such as the above-mentioned mutual

information and KL divergence, the classical kNN estimator has been also adopted to estimate Mutual Information (e.g. Kraskov et al., 2004) and KL divergence (e.g. Pérez-Cruz, 2008; Wang et al., 2009) nonparametrically when the underlying probability distribution is unknown.

However, one problem of the classical kNN estimator is that the uniformity assumption doesn't always hold especially when dimensionality grows. As a result, the kNN estimation of entropy has been found biased especially in higher dimensional cases (e.g. Noh et al; 2014).

Note that the $d\varepsilon$ is cancelled out from equation (2.2.2) to (2.2.5), which implies ono can replace the ε-ball with any shape which is controlled by some set of parameters $\boldsymbol{\varepsilon} = \{\varepsilon_1, \varepsilon_2, \varepsilon_3, \dots\}$ where the data points are assumed uniformly distributed in side of the defined hyper-region. It can also imply that the uniformity of the data points inside of the region influences the performance of the estimator.

2.3 Previous Work on Bias Reduction

The bias of the kNN estimator has attracted much attention. One group of approaches is to approximate the bias analytically and the other group of approaches solve the problem by conducting a shape correction locally.

Noh et al. (2014), in the context of estimating the KL divergence, have analytically approximated the bias by using a Taylor expansion of the probability density of the nearest neighbor $p(\boldsymbol{x}_i^{NN})$ around $\boldsymbol{x}_i$, so that

$$p(\boldsymbol{x}_i^{NN}) \approx p(\boldsymbol{x}_i) + \nabla p(\boldsymbol{x}_i)^T(\boldsymbol{x}_i^{NN} - \boldsymbol{x}_i) + \frac{1}{2}(\boldsymbol{x}_i^{NN} - \boldsymbol{x}_i)^T \nabla\nabla p(\boldsymbol{x}_i)(\boldsymbol{x}_i^{NN} - \boldsymbol{x}_i)$$

where $\nabla\nabla p(\boldsymbol{x}_i)$ denotes the Hessian of $p(\boldsymbol{x}_i)$.

Through complicated derivation, they obtained the bias of the estimated probability estimation as

$$Bias[\hat{p}(\boldsymbol{x}_i)] = E[p(\boldsymbol{x}_i^{NN})] - p(\boldsymbol{x}_i) \approx \alpha \nabla^2 p(\boldsymbol{x}_i)$$

and

$$\alpha = \frac{1}{2D(\gamma N p(\boldsymbol{x}_i))^{2/D}}$$

where $\nabla^2 p(\boldsymbol{x}_i) = tr(\nabla\nabla p(\boldsymbol{x}_i))$ denotes the Laplacian of $p(\boldsymbol{x}_i)$. When estimating the KL divergence, they further apply the approximation $\log(1+s) \approx s$, which yields

$$\begin{aligned}\log\frac{\hat{p}(\boldsymbol{x}_i)}{\hat{q}(\boldsymbol{x}_i)} &= \log\frac{p(\boldsymbol{x}_i)+\alpha_p\nabla^2 p(\boldsymbol{x}_i)}{q(\boldsymbol{x}_i)+\alpha_q\nabla^2 q(\boldsymbol{x}_i)} \\ &= \log\frac{p(\boldsymbol{x}_i)}{q(\boldsymbol{x}_i)} + \left(\log\frac{p(\boldsymbol{x}_i)+\alpha_p\nabla^2 p(\boldsymbol{x}_i)}{p(\boldsymbol{x}_i)} - \log\frac{q(\boldsymbol{x}_i)+\alpha_q\nabla^2 q(\boldsymbol{x}_i)}{q(\boldsymbol{x}_i)}\right) \\ &\approx \log\frac{p(\boldsymbol{x}_i)}{q(\boldsymbol{x}_i)} + \left(\log\alpha_p\nabla^2 p(\boldsymbol{x}_i) - \log\alpha_q\nabla^2 q(\boldsymbol{x}_i)\right)\end{aligned} \tag{2.3.1}$$

Thus, the bias of the KL divergence estimation is obtained

$$Bias\left[\log\frac{\hat{p}(\boldsymbol{x}_i)}{\hat{q}(\boldsymbol{x}_i)}\right] \approx \log\alpha_p\nabla^2 p(\boldsymbol{x}_i) - \log\alpha_q\nabla^2 q(\boldsymbol{x}_i).$$

After approximating the bias term, they have developed a metric learning approach to reduce the bias. The distance measure between $\boldsymbol{x}_i$ and $\boldsymbol{x}_i^{NN}$ is redefined as a Mahalanobis distance with a real-valued symmetric matrix A, such as

$$d(\boldsymbol{x}_i, \boldsymbol{x}_i^{NN}) = \sqrt{(\boldsymbol{x}_i^{NN} - \boldsymbol{x}_i)^T \mathrm{A} (\boldsymbol{x}_i^{NN} - \boldsymbol{x}_i)} \tag{2.3.2}$$

and the matrix A is learned via a semidefinite program

$$\min\,(tr[A^{-1}B])^2 \tag{2.3.3}$$

where

$$B = \alpha_p\frac{\nabla\nabla p(\boldsymbol{x}_i)}{p(\boldsymbol{x}_i)} - \alpha_q\frac{\nabla\nabla q(\boldsymbol{x}_i)}{q(\boldsymbol{x}_i)}. \tag{2.3.4}$$

They claimed that the matrix A which minimizes the $(tr[A^{-1}B])^2$ can minimize the bias and the solution to A is

$$A = \beta[U_+ \quad U_-]\begin{pmatrix} d_+\Lambda_+ & 0 \\ 0 & d_-\Lambda_- \end{pmatrix}[U_+ \quad U_-]^T \tag{2.3.5}$$

where Λ_+ and Λ_- are diagonal matrices contains positive and negative eigenvalues of the matrix B whereas d_+ and d_- are the corresponding numbers of the eigenvalues; U_+ and U_- are consist of eigenvectors corresponding to Λ_+ and Λ_-. Since the probability density function and the Hessian are unknown, Gaussian models are applied to obtain the matrix B, so that the first term of B in (2.3.4) can be obtained via

$$\frac{\nabla\nabla p(\boldsymbol{x}_i)}{p(\boldsymbol{x}_i)} = \hat{\Sigma}^{-1}(\boldsymbol{x}_i - \hat{\mu})(\boldsymbol{x}_i - \hat{\mu})^T\hat{\Sigma}^{-1} - \hat{\Sigma}^{-1}$$

where $\hat{\mu}$ and $\hat{\Sigma}$ are mean vector and covariance matrix of the whole data obtained by maximum-likelihood estimation, and the second term can be handled similarly.

Note that, in theory, the notion should be also applicable in entropy estimation, similar to equation (2.3.1), the bias term for the entropy alone can be obtained via

$$\begin{aligned}\log\hat{p}(\boldsymbol{x}_i) &= \log\left(p(\boldsymbol{x}_i) + \alpha_p\nabla^2 p(\boldsymbol{x}_i)\right) = \log p(\boldsymbol{x}_i) + \log\left(\frac{p(\boldsymbol{x}_i) + \alpha_p\nabla^2 p(\boldsymbol{x}_i)}{p(\boldsymbol{x}_i)}\right) \\ &\approx \log p(\boldsymbol{x}_i) + \log\alpha_p\nabla^2 p(\boldsymbol{x}_i).\end{aligned}$$

Now the bias-reduction becomes a metric learning process similar to the above-mentioned procedure but with a different B matrix (only the first term remains). The procedure becomes solving the semidefinite program from equation (2.3.3) to (2.3.5) with a difference matrix B where

$$B = \alpha_p\frac{\nabla\nabla p(\boldsymbol{x}_i)}{p(\boldsymbol{x}_i)}.$$

Analytically approximating bias via a Taylor expansion can also be found in literature of related other nonparametric methods such as kernel density estimation (KDE) (e.g. Calonico and Cattaneo,

2015). For example, in Calonico and Cattaneo's work, similar to Noh et al.'s work, the bias is obtained by a second-order Taylor expansion and is expressed in the form of a Laplacian function. However, one problem of that approach is that the bias is expressed as a degree of the curvature (Laplacian) of the underlying probability distribution, however, in some cases, it is possible that the underlying distribution is even more complicated so that the complexity is not well described by the curvature alone.

The other issue in Noh et al.'s (2014) approach is that there are some potentially unrealistic approximations. One is the $\log(1+s) \approx s$ approximation used while obtaining the bias. The similarity only holds when s is a very small number. Besides, the s which replaces of the $\log(1+s)$ is in fact the upper bound of $\log(1+s)$ since

$$\frac{s}{1+s} \leq \log(1+s) \leq s.$$

Therefore, the usage of the similarity approximation results in an over estimation of the bias. The other one is the Gaussian approximation of the underlying distribution. This assumption can be unrealistic in some real-world cases especially when the underlying distribution is asymmetric. Recently, Sasaki et al. (2016) have proposed a novel approach based on Noh et.al's (2014) work to estimate KL divergence; the approach nonparametrically estimates the Hessian and density ratio without assuming any underlying distribution, however, Sasaki et al.'s method can be only applied to KL divergence estimation and not to entropy estimation.

Another approach is to relax the uniformity assumption (Gao et al., 2015; Lord et al., 2017). Instead of approximating the bias term, this approach assigns another local shape which is believed to hold more uniformly distributed data points inside. Gao et al.'s method, although it doesn't focus on a kNN estimator but another estimator called KSG estimator (Kraskov et al., 2004), has been one of the representative examples. The KSG estimator is similar to a kNN estimator but unlike a kNN estimator, the KSG estimator utilizes the max-norm distance and it assumes a uniformly distributed hypercube instead of the Euclidean distance and uniformly distributed ε-hypersphere used in a kNN estimator.

Gao et al. therefore assume that instead of having uniformity inside the V_i^D-volumed ε-hypercube, the uniformity holds in a subset of the ε-hypercube, which is a hypercube with the volume $\bar{V}_i^D$. The local nonuniformity correction (LCN) term $\widehat{H}_{LNC}(\boldsymbol{X})$ is

$$\widehat{H}_{LNC}(\boldsymbol{X}) = \widehat{H}_{KSG}(\boldsymbol{X}) - \frac{1}{N}\sum_{i=1}^{N} \log \frac{\bar{V}_i^D}{V_i^D}$$

where $\widehat{H}_{KSG}(\boldsymbol{X})$ denotes the KSG estimator and the the $\bar{V}_i^D$ is learned via performing a local principle component analysis (PCA).

Gao et al. have further introduced a test procedure to avoid over-correction. For each $\boldsymbol{x}_i$, the correction term $\log \frac{\bar{V}_i^D}{V_i^D}$ has to be smaller than a pre-defined constant $\alpha_{k,D}$; if the correction term is not smaller than $\alpha_{k,D}$, the correction is considered not necessary and discarded. The $\alpha_{k,D}$ is determined by sampling N (Gao et al. suggest 5×10^5) sample sets from a D-dimensional uniform distribution, then calculating the correction term $\frac{\bar{V}_i^D}{V_i^D}$ of each sample set, and then selecting the ϵN-th smallest $\frac{\bar{V}_i^D}{V_i^D}$ as the constant $\alpha_{k,D}$, where ϵ is a small probability (Gao et al. suggest 5×10^{-3}).

As mentioned before, Gao et al.'s work focuses on the KSG estimator, and the same notion of adjustment hasn't been successfully done on kNN estimators. Besides, the determination of the testing constant $\alpha_{k,D}$ is arbitrary, the choice of N as well as ϵ can influence the selected constant $\alpha_{k,D}$ and furthermore influence the quality of the correction.

2.4 MCMC posterior samples evaluation

In Bayesian inference, the form of the posterior distribution can sometimes be unknown or not analytically computable especially in complicated model settings. The simulation approach such as Markov chain Monte-Carlo (MCMC), or Approximate Bayesian Computation (ABC) enable researchers to circumvent derivation of the posterior form directly but allow simulating samples from the posterior for inference.

In sampling-based inference, the ability of the posterior samples to reveal the features of the true posterior is surely important. Biased posterior samples can lead to inaccurate inference influencing decision making.

This problem has been considered as a convergence assessment task and many approaches have been proposed (e.g. Gelman and Rubin, 1992; Roberts, 1992, 1994). One group of methods focus on assessing whether the samples are representative of the underlying stationary distribution (Cowles and Carlin, 1996). For more detailed information, Cowles and Carlin (Cowles and Carlin, 1996) and El Adlouni et al. (El Adlouni et al., 2006) have made systematic review articles.

One instance of a convergence assessment approach is Gelman and Rubin's method, where they have proposed monitoring a shrinkage factor based on running m parallel MCM chains e.g., from different initializations and computing a statistic defined as

$$\sqrt{R} = \sqrt{\left(\frac{n-1}{n} + \frac{m+1}{mn}\frac{B}{W}\right)\frac{df}{df-2}}$$

where n is the number of iterations so far, B is the variance between the means of the m chains and W is the average with-in variances of the m chains. Gelman and Rubin suggest that the samples become better converged as the $\sqrt{R}$ shrinks to one.

Another example is Robert's approach (Roberts, 1992, 1994) which is based on the regularity condition that

$$\left\| f^{(n)} - f \right\| \xrightarrow{n\to\infty} 0$$

where f is the targeted distribution, $f^{(n)}$ the distribution at the n-th iteration and $\| \, . \, \|$ is a norm related to a particular inner product. The monitor procedure is then developed, by defining:

$$\chi_n^{lp} = \frac{k(\theta_l^{(0)}, \theta_p^{(n)})}{f(\theta^{(2n-1)})}$$

$$D_n = \frac{1}{m}\sum_{l=1}^{m} \chi_n^{ll}$$

$$I_n = \frac{1}{m(m-1)}\sum_{l \neq p} \chi_n^{lp}$$

where k is a backward kernel in a Gibb's sampler, l and p are indices of different samplers and $\theta_p^{(n)}$ denotes the obtained value after n iterations from the pth chain. D_n is defined as dependent term and I_n is defined as the interactive term. Roberts showed that $E[D_n] = E[I_n]$ when the chain has been converged. He then suggests using $m = 10$ to 20 initializations and monitoring the values of D_n and I_n until they are close enough.

One potential issue with these kinds of approaches is that they cannot detect if the samples are converging to another incorrect probability distribution instead of the true posterior distribution. Another issue with these kinds of approaches is that some of the approaches are often limited to certain types of samplers, for example, Raftery and Lewis's approach (Raftery and Lewis, 1992) is specific to Gibbs samplers and Robert's approach is limited to symmetric Gibbs samplers. This limitation restricts the potential of comparing the performances of different samplers together.

The other group of approaches for evaluation of the posterior samples (Chauveau and Vandekerkhove 2014; Gorham and Mackey, 2015; Cusumano-Towner and Mansinghka, 2016; Grosse et al, 2016) solve the problem by approximaing the divergence or deviance between the samples and the true posterior distribution.

Gorham and Mackey (2015) have proposed an approach to estimate the maximum deviance using Stein's method. However, in the one of the simulation experiments in this thesis, this approach, probably due to their choice of the measurement (maximum deviance), it has been found poor discriminative with a simple univariate experiment which can be found in the Section 4; in detail it requires more samples to discern which of the two samplers is better than a divergence-based approach does.

Grosse et al. (2016) have also proposed a framework to estimate the symmetric KL-divergence between the posterior samples and the true posterior. Grosse et al.'s approach is specific for AIS

(Annealed Slice Sampler, Neal, 2001) based samplers, therefore, the application of the approach is limited.

Cusumano-Towner and Mansinghka's (2016) work also focuses on the symmetric KL divergence between the "golden standard" oracle sampler and the evaluated sampler. However, an oracle sampler in real-world cases is not always available, in fact, in most of the cases, the true posterior or the best sampler is unknown. Therefore, the application of this approach is again restricted.

Chauveau and Vandekerkhov (2014) have proposed an approach measure the KL divergence from the true posterior to the posterior samples. However, the porposed approach utilize the classical kNN for the entropy estimation which makes the evaluation suffer from the bias issues in high-dimensional cases. More details of their work and the proposed improvement will be further discussed in Section 3.4.

3. Method

In this section, the kNN estimator with Ellipsoidal correction (EC-kNN) is developed. The algorithm comprises two parts, one is the local Ellipsoidal correction where a local ellipsoid is learned via performing a local PCA algorithm, the other part is the acceptance testing procedure which is performed in a boot-strapping manner.

The local ellipsoid is approximated via performing a local PCA using the neighbors of the interested sample point. After performing the local PCA, the ratios of axes are utilized for the local volume correction.

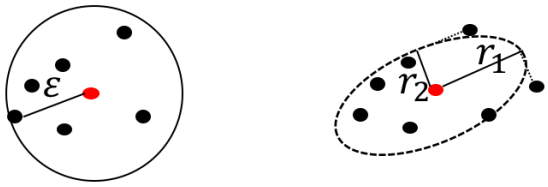
On the other hand, the acceptance procedure simulates samples from a uniform distribution (assumed by the classical kNN estimator) with the same setting of N samples and the number of neighbors set to k in order to compute a corresponding correction for the random sample. The acceptance is determined based on whether the observed correction is greater than the random generated correction.

3.1 Ellipsoidal Correction

Here a correction is presented for the bias caused from uniformity assumption based on ellipsoidal approximation. The basic idea is to construct a local ellipsoid so that inside of it data are assumed to be more uniformly distributed than inside local ball-shaped structure.

For learning the local ellipsoid structure, the local PCA algorithm is adopted in this work to learn the local ellipsoidal structure around the sample of interest point $\boldsymbol{x}_i$. The local PCA first computes the covariance matrix of the neighborhood of $\boldsymbol{x}_i$ ($\boldsymbol{x}_i$ is included) and rotate the neighborhood to a new coordinate system by projecting the data onto the eigenvectors of the obtained covariance matrix. The axes of the local ellipsoid are then computed via searching for the maximum distance along each coordinate axis.

After performing the local PCA, the sum of the log of ratios of the longest axis to other each of the axes of the estimated ellipsoid is then taken as the logarithmic volume correction term; additionally, the number of neighbors is recounted based on the ellipsoid.



(a) (b)

Figure 2.1: (a) The ε-ball of the classical kNN estimator. (b) Local ellipsoid learned via local PCA

Since the volume of an ellipsoid with axes $r_1, r_2, \ldots, r_d$ is defined as

$$\frac{4}{3}\pi \prod_{d=1}^{D} r_d$$

, we assume that the longest axis r_1, the distance from the origin to the farthest data point alone the longest axis, represents the original radius, thus, the correction term can be obtained as

$$\Delta\hat{V}(\boldsymbol{x}_i, \boldsymbol{X}) = \frac{{r_1}^D}{\prod_{d=1}^{D} r_d} = \prod_{d=1}^{D}(r_1/r_d).$$

An alternative option can be using the original radius of the hypersphere (ε), however, we found that option to have poor performance in simulation experiments.

Note that, in high dimensional cases, it can happen that when the number of neighbors inside the new ellipsoid are counted, it turns out that in that there are no points inside of the ellipsoid. When encountering this issue, the axes of the ellipsoid are increased slightly until there is at least one data point inside of the ellipsoid. Here, in the proposed algorithm, every one of the axes is lengthened by multiplying the r_d by a small ratio (e.g., 1.01). The volume correction is changed accordingly.

By definition, one of the points $\boldsymbol{x}_i = [x_{i,2}, x_{i,2}, \ldots x_{i,d}, \ldots x_{i,D}]$ is inside of the ellipsoid if

$$\sum_{d=1}^{D} \frac{(x_{i,d} - c_d)^2}{{r_d}^2} \leq 1$$

where $\boldsymbol{c} = [c_1, c_2 \ldots c_d, \ldots c_D]$ denotes for origin of the ellipsoid.

Algorithm 1: local ellipsoid-based volume correction

Input:

$\boldsymbol{x}_i$: sample point

$\boldsymbol{X} = \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_i, \ldots \boldsymbol{x}_N$: other sample points

D: dimension

k: number of neighbor points

Output: $\Delta\hat{V}_k(\boldsymbol{x}_i, \boldsymbol{X})$: local ellipsoid-based volume correction

Find kth nearest neighbors of $\boldsymbol{x}_i$ (typically by Euclidean distance), get the distance ε_i to the kth neighbor sample

Perform a PCA on the set of $k + 1$ points including the its k neighbors and $\boldsymbol{x}_i$. Then project $\boldsymbol{x}_i$ and its k neighbors to the new coordinate system. Get the new center by averaging all the projected points.

Find the lengths $r_1, r_2, \ldots r_D$ of the axes of the projected ellipsoid through computing the maximum difference of data points from the center along each PCA projection axis.

While there are no points inside of the ellipsoid **Do**

lengthen every axis by multiplying each r_d with a small ratio (e.g., 1.01)

Until at least one point is inside of the ellipsoid

Compute the volume changing ratio

$$\Delta\hat{V}(\boldsymbol{x}_i, \boldsymbol{X}) = \prod_{d=1}^{D}(r_1/r_d)$$

3.2 Bootstrap test of correction acceptance

Due to the fact that the nonuniform distribution of data along different coordinate axes can also happen due to of the random sample variation, that is, the nonuniform distribution of a particular data subset can be observed even under the uniformity assumption, it can happen that the corrections from some points are not necessary and the over-correction problem occurs if the correction is performed in every sample point.

To address this problem, an acceptance procedure is introduced. For each sample point, an auxiliary variable $\hat{V}_u$ is generated as an acceptance variable to amend the potential over-correction issue. The variable is simply a volume correction term obtained from a set of $k+1$ samples generated inside of a uniformly distributed ε-ball around the $\boldsymbol{x}_i$. The details of generating the $\hat{V}_u$ is shonwn in Algorithm 2.

Since the data point $\boldsymbol{x}_i$ is itself randomly generated, in order to simulate a true random configuration, the randomness of that the $\boldsymbol{x}_i$ is the center of the ε-ball should also be taken into account. Therefore, after simulating the $k+1$ points inside of the ε-ball, the center of the ball is not necessary the $\boldsymbol{x}_i$but redefined by choosing the point which is closest to the original point. The correction term for the $k+1$ points is then computed by the Algorithm 1. If the originally obtained correction $\tilde{V}_i$ is smaller than $\hat{V}_u$, the twist around the sample point is considered to likely be due to randomness, and the correction $\tilde{V}_i$ is discarded. Note that the role that the variable $\hat{V}_u$ plays is similar to the $\alpha_{k,d}$ parameter in Gao's work, but instead of an arbitrary, pre-assigned parameter, the variable $\hat{V}_u$in the proposed algorithm is determined by sampling based on the observed data.

Algorithm 2: Acceptance variable

Input:

D: dimension

k: number of neighbor points

ε: the radius

Output: $\hat{V}_u$: acceptance variable

Generate random samples $\boldsymbol{U} = \boldsymbol{u}_1, \boldsymbol{u}_2, \dots, \boldsymbol{u}_{k+1}$ from a D-dimensional uniformly distributed ε-ball.

Take the point $\boldsymbol{u}_j$ which is closest to the original among $\boldsymbol{U}$ as the new center.

Compute the correction term $\hat{V}_u = \Delta\hat{V}(\boldsymbol{u}_j, \boldsymbol{U})$

3.3 KNN estimator with ellipsoidal correction

The proposed kNN estimator with ellipsoidal correction (EC-kNN) is simply a combination of the above-proposed algorithms. In the proposed algorithm, for each sample point $\boldsymbol{x}_i$, a local ellipsoidal correction is performed with a bootstrap acceptance test. The correction term $\tilde{V}_i$ and the referenced correction term $\hat{V}_u$ are generated by Algorithm 1. and Algorithm 2. Respectively. Then the bootstrap acceptance test is conducted via comparing the values of $\tilde{V}_i$ and $\hat{V}_u$. The correction Is accepted if $\tilde{V}_i < \hat{V}_u$, otherwise the algorithm uses the result of the classical kNN estimator.

After going through every data point in the data set, the algorithm produces the final corrected result by averaging the corrected entropies from each data point from the data set. Note that, since the bootstrap acceptance test procedure is a result of a randomly generated value $\hat{V}_u$, therefore, the result of the computation can be slightly different every time.

Algorithm 3: EC-kNN

Input:

$\boldsymbol{X} = \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$: samples

D: dimension

k: number of neighbor points

Output: $\widehat{H}(\boldsymbol{X})$: corrected entropy estimation

for each $\boldsymbol{x}_i$ do

Find its k neighbors, get the distance ϵ_i to the kth neighbor sample and compute volume of the ε_i-ball

$$V_i = \frac{\pi^{\frac{D}{2}}}{\Gamma(1+\frac{D}{2})}.$$

Compute the reference volume correction $\tilde{V}_i = \Delta\hat{V}_k(\boldsymbol{x}_i, \boldsymbol{X})$

Generate the acceptance variable $\hat{V}_u$

If $\tilde{V}_i < \hat{V}_u$ **do**

Find the number of the neighbor points k_i inside of the projected ellipsoid

$$H(\boldsymbol{x}_i) = \psi(N) - \psi(k_i) + \log(V_i) + \log(\tilde{V}_i)$$

else

$$H(\boldsymbol{x}_i) = \psi(N) - \psi(k) + \log(V_i)$$

end for

$$\widehat{H}(\boldsymbol{X}) = \frac{1}{N}\sum_{i=1}^{N} H(\boldsymbol{x}_i)$$

3.4 Divergence-based evaluation of MCMC posterior samples with EC-kNN

Let us take a closer look at the KL divergence. Let θ denote the parameter and y denote the data. In general, Bayesian inference aims to find the posterior distribution,

$$\pi(\theta|y) = \frac{\pi(\theta)p(y|\theta)}{p(y)}$$

which combines the information of the prior distribution $\pi(\theta)$ and the likelihood $p(y|\theta)$. The normalizing constant $p(y)$ can be obtained by

$$p(y) = \int \pi(\theta)p(y|\theta)d\,\theta.$$

Since the exact form of the posterior $\pi(\theta|y)$ is not always available especially when dealing with complicated models, in practice, an approximating distribution $q(\theta)$ is employed to circumvent deriving the exact form of the posterior distribution. The approximating distribution $q(\theta)$ can be obtained by simulation tools such as MCMC (Monte Carlo Markov Chain) or approximation techniques such as Variational Bayes method

The KL from the true posterior distribution $\pi(\theta|y)$ to the approximating distribution $q(\theta)$ can be written as

$$\begin{aligned} &D_{KL}\big(q(\theta), \pi(\theta|y)\big) \\ &\qquad = \Big(D_{KL}\big(q(\theta), \pi(\theta)\big) - D_{KL}\big(\pi(\theta|y), \pi(\theta)\big)\Big) \\ &\qquad - \Big(\Big[E_{q(\theta)}[\log p(y|\theta)] - E_{\pi(\theta|y)}[\log p(y|\theta)]\Big]\Big) \end{aligned}$$

where the divergence between the true and approximated posterior distribution has been decomposed into the difference of their own Kullback-Leibler divergence to the prior distribution $\pi(\theta)$ and the difference of the expected log-likelihood. The KL divergence to the prior distribution is also called Bayesian surprise (Itti and Baldi, 2005). The above divergence can also be written as

$$\begin{aligned} &D_{KL}\big(q(\theta), \pi(\theta)\big) - E_{q(\theta)}[\log p(y|\theta)] - E_{\pi(\theta)}[p(y|\theta)] \\ &\qquad = H\big(q(\theta)\big) - E_{q(\theta)}[\log \pi(\theta)p(y|\theta)] - E_{\pi(\theta)}[p(y|\theta)] \end{aligned}$$

where $E_{\pi(\theta)}[p(y|\theta)]$ is the normalizing constant which is always greater than zero; the value of this term can be obtained by techniques such as annealed importance sampling (AIS, Neal, 2001), however, since the term has nothing to do with the $q(\theta)$, when comparing two different samplers, the term can be ignored.

The negative of the term

$$D_{KL}\big(q(\theta), \pi(\theta)\big) - E_{q(\theta)}[\log p(y|\theta)]$$

is also called the evidence lower bound (ELBO) in variational Bayes (Blei et al., 2016), and the variational Bayes inference algorithm is aiming at optimizing the ELBO value. In Chauveau and Vandekerkhove's (2014) work, the term

$$H\big(q(\theta)\big) - E_{q(\theta)}[\log \pi(\theta) p(y|\theta)] - E_{\pi(\theta)}[p(y|\theta)] \qquad (3.4.1)$$

is calculated to evaluate the MCMC posterior samples. Chauveau and Vandekerkhov have used the classical kNN estimator to estimate the entropy $H\big(q(\theta)\big)$ from the posterior sample and then have used Monte-Carlo intergration to obtain the term $E_{q(\theta)}[\pi(\theta)\log p(y|\theta)]$. However, they haven't dealt with the above-mentioned bias issue of the classical kNN estimator, therefore, the applicability of their convergence assessment method is limited to low dimensional cases.

In this thesis, we then suggest using the proposed EC-kNN entropy estimator to estimate the value of $H\big(q(\theta)\big)$ in the equation (3.4.1). Using the EC-kNN can surely address the bias issue caused by the classical kNN and enhance the correctness when evaluating the quality (deviance from the true posterior distribution) of MCMC posterior samples evaluation. In the Section 5., we demonstrate working examples and clarify the improvement.

4. Simulation Study

To verify the usability of the proposed approach and to demonstrate that the proposed approach can really contribute to the research community, a simulation study is conducted in a comparative manner. The naïve kNN estimator is used as the baseline approach. The method proposed by Noh et al. (2015) is also selected to compare with the proposed algorithm. For the naïve kNN estimator (kNN), Noh et al.'s (2014) estimator (Noh) and the proposed estimator (EC-kNN), we fix the number of the neighbors to be $k = 25$.

4.1 Symmetric Gaussian Case

In this simulation task a multivariate Gaussian distribution is selected to be the ground truth distribution, because the entropy $H(\boldsymbol{X})$ of a multivariate Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ can be analytically obtained. When

$$\boldsymbol{X} \sim \boldsymbol{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}).$$

The corresponding entropy is

$$H(\boldsymbol{X}) = \frac{1}{2} \log \det(2\pi e \boldsymbol{\Sigma}).$$

Since the entropy $H(\boldsymbol{X})$ does not depend on the value of the mean vector $\boldsymbol{\mu}$, it is then fixed to a zero vector, so that

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}.$$

In this symmetric Gaussian case, the covariance is designed as

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0 & & \\ 0 & 1 & & \\ & & \ddots & 0 \\ & & 0 & 1 \end{bmatrix}$$

where the variance of each dimension is fixed to 1 and the increase of the dimensionality doesn't influence the symmetry of the distribution.

While conducting the experiment, for each iteration, 500 data points are sampled from the distribution to estimate the entropy and 10 iterations are repeated for each dimension. For each iteration the estimated entropy is compared to the true entropy and the squared error between the estimate and the true value is computed. The root mean of the 10 squared error values (RMSE) over the iterations is used to compare the performances of different approaches.

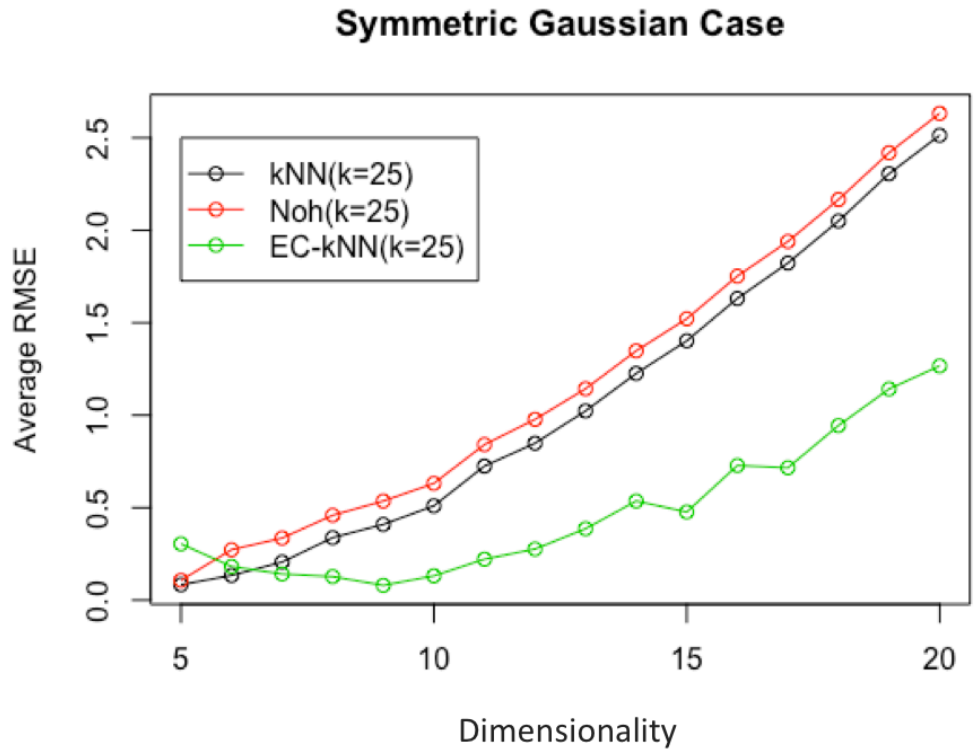


Figure 4.1. Performance comparison in symmetric mixture case. The EC-kNN (green) outperforms kNN (black) and Noh et al.'s approach (red).

As shown in Figure 4.1. The RMSE values grow as the dimensionality becomes higher. The performances of the three approaches are similar with relatively small RMSE in lower dimensions (from 5 to 10), however, as the dimension grows, the proposed EC-kNN estimator starts to outperform other approaches. The approach of Noh et al. (2015) performs the worst among all approaches.

4.2 Asymmetric Gaussian Case

Following the symmetric case, an experiment on a more complicated asymmetric Gaussian case is provided here. In this case, the mean vector is still fixed to a zero vector

$$\boldsymbol{\mu} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

but unlike the previous case, the values in the diagonal of the covariance matrix increases as the dimensionality grows, so that for dimensionality D we set.

$$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0 & & \\ 0 & 2 & & \\ & & \ddots & 0 \\ & & 0 & D \end{bmatrix}.$$

Like the previous case, 500 data points are sampled for each experiment and the experiment repeats 10 times in each dimension and the RMSE is taken to compare the performances.

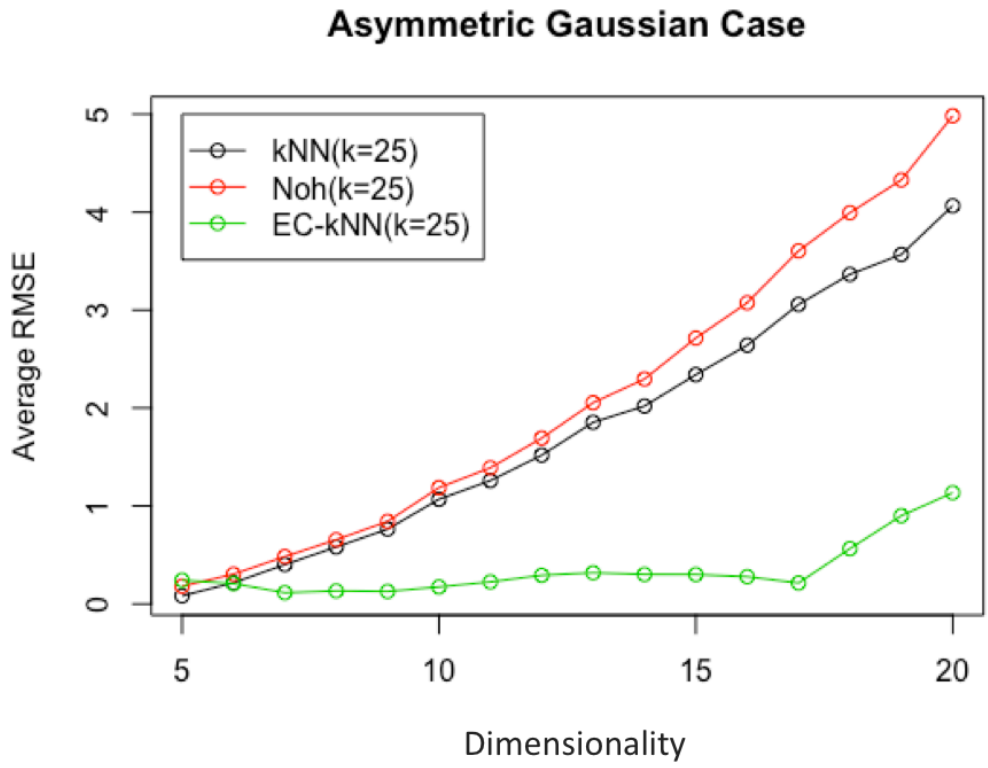


Figure 4.2. Performance compared in asymmetric mixture case. The EC-kNN (green) outperforms kNN (black) and Noh et al.'s approach (red).

As shown in Figure 4.2., compared to the previous case, the RMSE value is bigger when the underlying distribution is asymmetric and the advantage of the proposed EC-kNN algorithm is more obvious, especially in higher dimensional spaces. The approach of Noh et al. (2015) still suffers from over-correction.

4.3 Mixture Gaussian Case

In the third simulation experiment, a mixture of two Gaussian distributions is selected to evaluate the performance of the proposed method in a scenario which is more complicated than the previous two cases. The probability density function of a mixture of two Gaussian distributions is defined as

$$p(x) = \pi p(\boldsymbol{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) + (1-\pi)p(\boldsymbol{x}|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2).$$

where $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ are mean vectors and $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$ are covariance matrices of the two Gaussian distributions and the π is the mixture weight.

The parameter setting is as follows

$$\pi = 0.4$$

$$\boldsymbol{\mu}_1 = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \ \boldsymbol{\Sigma}_1 = \begin{bmatrix} 1 & 0 & & \\ 0 & 2 & & \\ & & \ddots & 0 \\ & & 0 & D \end{bmatrix},$$

$$\boldsymbol{\mu}_2 = \begin{bmatrix} 10 \\ \vdots \\ 10 \end{bmatrix}, \boldsymbol{\Sigma}_2 = \begin{bmatrix} 1 & 0 & & \\ 0 & 1 & & \\ & & \ddots & 0 \\ & & 0 & 1 \end{bmatrix}.$$

The above-defined distribution is a combination of two different Gaussian distributions, where one of them is an asymmetric Gaussian distribution and the other one is a symmetric Gaussian distribution.

Since the entropy of a mixture Gaussian distribution cannot be obtained analytically, for each dimension, 100000 samples are simulated from the designed distribution and are taken to estimate the entropy through a Monte-Carlo integration, so that

$$\hat{H}(\boldsymbol{X}) = \int -p(\boldsymbol{x})\log p(\boldsymbol{x})\, dx \approx -\frac{1}{N}\sum_{i=1}^{N} \log p(\boldsymbol{x}_i).$$

The result of the Monte-Carlo integration is then taken to represent the true value of the entropy. Like the previous experiments, 500 data points are generated in each iteration and for each dimension, 10 iterations are conducted, the average of the 10 RMSE values is still taken as the performance measurement.

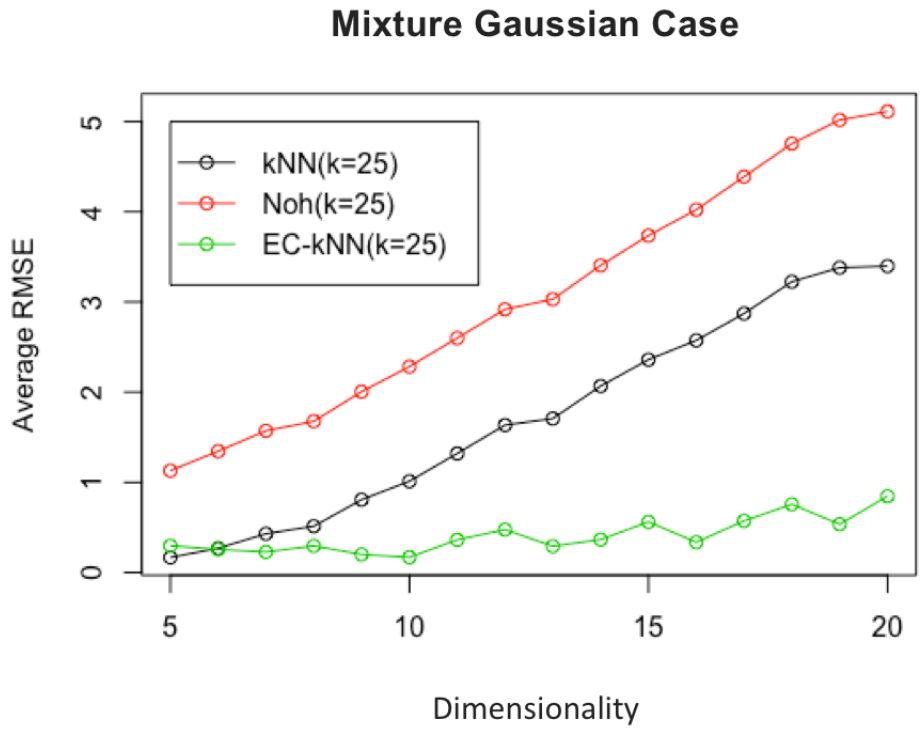


Figure 4.3. Performance comparison in Mixture Gaussian Case. The EC-kNN (green) outperforms kNN (black) and Noh et al.'s approach (red).

As shown in Figure 4.3., the result in the mixture Gaussian case is similar to the two previous cases. The approach of Noh et. al. (2015) has the poorest performance and the proposed approach outperforms other approach, which implies that it is suitable when the underlying distribution is complicated.

In the previous 3 cases, the EC-kNN outperforms the other approaches. Note that both the naïve kNN and Noh's method are sensitive to the dimensionality, their error increases while the dimensionality grows whereas the performance of EC-kNN is relatively stable when the dimension grows.

5. Application to MCMC Posterior Samples Evaluation

5.1 Application Example 1. Univariate Normal-Normal conjugate case

In this section, a comparison is done between two approaches to estimate divergence between MCMC samples and the true posterior distribution, the approach of Gorham and Machey (2015) and the divergence-based approach. The simulation task is conducted based on a univariate Normal-Normal conjugate example.

The model setting is as follows

$$Y_i \sim N(\theta, 10)$$
$$\theta \sim N(0, 10)$$

and the observed data consists 9 values with the average of zero. This is the case of a univariate Gaussian distribution with known variance and unknown mean with a conjugate prior. The posterior can be simply obtained following the known posterior equations for this case committed for brevity, and the result is

$$\theta | y \sim N(0,1)$$

Then assume there are two samplers in hand, one generates samples from a student-t distribution with $df = 20$ degree of freedom, zero mean and variance equals to $\frac{20}{18} \sim 1.11$, the other sampler generates samples from a Gaussian distribution with zero mean and variance equal to 1, in other words from the correct posterior distribution.

The method of Gorham and Mackey (2015) and the divergence-based method are further applied to compare the two samplers. Since this a is univariate example, the classical kNN estimator is utilized as in the univariate case the proposed EC-kNN estimator reduces to the classical kNN estimator.

As shown in Figure 5.1 and Figure 5.2, the Gorham et al.'s approach starts to distinguish the difference between two samplers while the sample size is greater than 2000 whereas the divergence-based approach requires only around 200 samples. This indicates that the latter approach is promising. Next we demonstrate a high-dimensional working example.

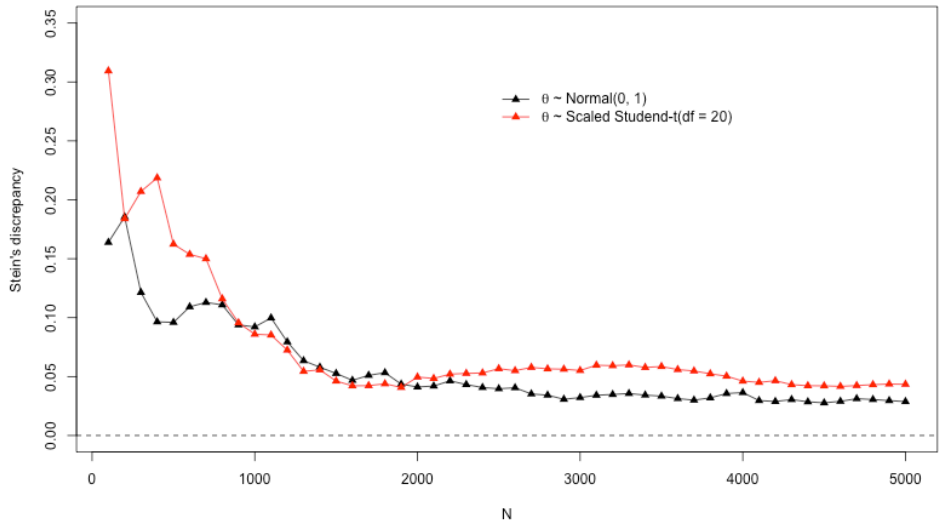


Figure 5.1: The approach of Gorham and Mackey (2015). No obvious difference between the samples from Normal distribution (black bots) and student-t distribution (red triangles) when sample size is small (less than 2000).

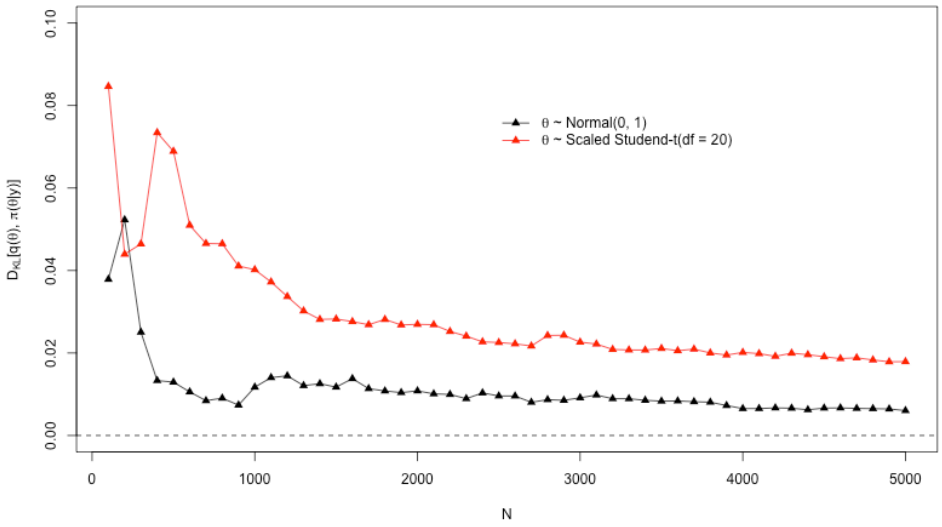


Figure 5.2: Divergence-based approach. Obvious difference between the samples from Normal distribution (black bots) and student-t distribution (red triangles) can be found in early iterations.

5.2 Application Example 2. Bayesian Linear Regression

Here a Bayesian linear regression example is demonstrated. The model setting is as follows:

$$Y_i \sim N(\boldsymbol{\beta}^T \boldsymbol{X}_i, 100)$$
$$\beta_j \sim N(0, \alpha)$$
$$\alpha \sim Gamma(10, 1)$$

where Y_i denotes the dependent variable, $\boldsymbol{X}_i$ is a vector of independent variables, the parameter vector $\boldsymbol{\beta}$ is a vector of the coefficients and the model noise, for simplicity, is fixed to 100.

The wage dataset from an R package statsr (Rundel et al.) is used, the dataset contains the wage of randomly sampled 935 respondents and also other 15 variables including working hours per week, IQ score and years of education and so on. In this Bayesian linear regression model, the wage variable is taken as the dependent variable, whereas the other 15 variables are taken as predictors $\boldsymbol{X}$**.** Therefore. there are 16 coefficients β_j (including the intercept) and the posterior distribuend is 17-dimensional (α is also included). The rows containing NA values are ruled out. In total, 663 complete cases are used for model building.

Two sampling strategies are evaluated, one is a pure Metropolis-Hasting sampler (MH sampler, Hasting, 1970) and the other one is a combination of the MH sampler and ESS (Elliptical Slice sampler; Murray, 2010). The Metropolis-Hasting sampler is a classical sampler, whereas the ESS is a sampler designed for Bayesian models with Gaussian priors.

To evaluate the performances of the above-mentioned samplers, two sampling algorithms are designed. The parameters are $\Theta = \{\alpha, \boldsymbol{\beta}\}$. For the first one, α is obtained with the MH algorithm with a Gaussian proposal distribution and the $\boldsymbol{\beta}$ is obtained with an elliptical slice sampler. For second algorithm, both α and $\boldsymbol{\beta}$ are obtained by the MH sampler with Gaussian proposal distributions

A better initialization is given to the second algorithm. The first sampler (the one which includes the ESS) is executed first and the 1000^{th} posterior sample is taken as the initialization for the second algorithm.

Both sampling algorithms run 10000 iterations. For each algorithm, the ELBO value

$$H\big(q(\theta)\big) - E_{q(\theta)}[\log \pi(\theta) p(y|\theta)]$$

is computed from the 701th iteration onwards. The kNN and EC-kNN estimators taking the previous 500 posterior samples are applied to estimate the entropy term $H\big(q(\theta)\big)$ and the term $E_{q(\theta)}[\pi(\theta)\log p(y|\theta)]$ is estimated via Monte-Carlo integration, that is,

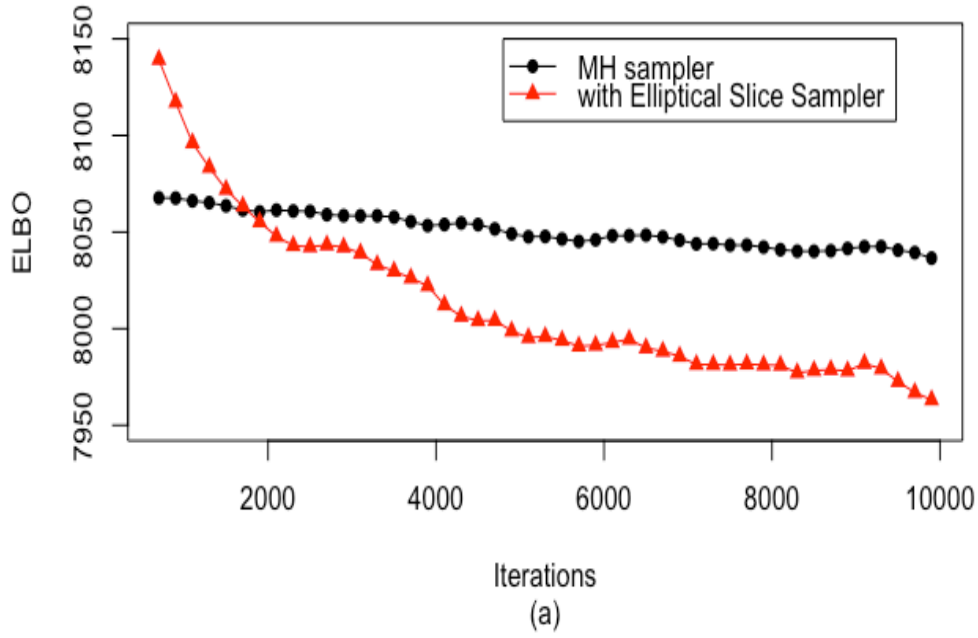
$$\frac{1}{500}\sum_t \log \pi\big(\theta^{(t)}\big) p\big(y\big|\theta^{(t)}\big)$$

where

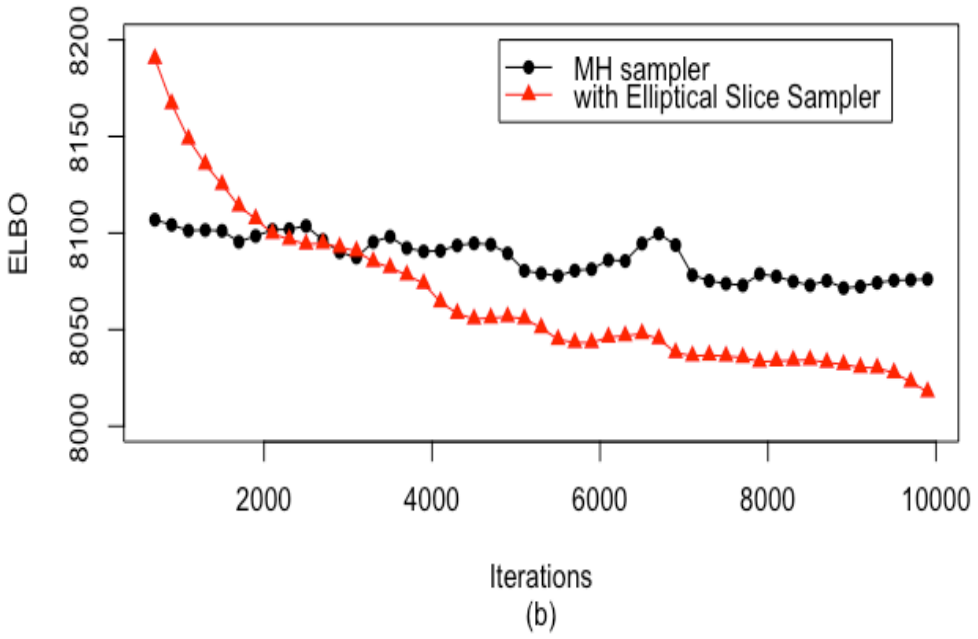
$$\pi\big(\theta^{(t)}\big) = Gamma(\alpha^{(t)}|10,1)\prod_j Normal(\beta_j^{(t)}|\alpha^{(t)})$$

$$p\big(y\big|\theta^{(t)}\big) = \prod_i Normal(y_i|\boldsymbol{\beta}^{(t)^T}\boldsymbol{x}_i{}^{(t)}, 100)$$

As Shown in Figure 4.3, although the algorithm with solely MH samplers is starting from a better initialization, the two types of ELBO estimations are leading to the same results, and they both favor the algorithm comprising the elliptical slice sampler. It has already been proven in the previous section that in general, the EC-kNN estimator is more accurate than the classical kNN estimator. Furthermore, in this case, the estimated ELBO computed via the EC-KNN estimator distinguishes the difference of the two algorithms in already earlier iterations (the improved performance of the ESS included algorithm is obvious after the 2000th iteration while the kNN based estimate reveals the difference only after the 3300th iteration).
This demonstrated working example shows the advantage of the EC-KNN and can be also a framework to evaluate other sampling algorithms.



(a)



(b)

Figure 5.3 (a): Performance comparison between two sampling algorithms using the EC-kNN estimated ELBO values. (b) Performance comparison between two sampling algorithms using the kNN estimated ELBO values. Sampler 1 (black dots) generates samples based on pure MH samplers whereas Sampler 2 (red triangles) generates α with a MH sampler but generates $\boldsymbol{\beta}$ with a elliptical slice sampler. Both ELBO evaluation methods show that Sampler 2 out performs Sampler 1, however, the EC-kNN estimated ELBO is more discriminative than teh kNN estimated ELBO since it distinguishes the two samplers in earlier iterations.

6. Discussion and Conclusion

The contribution of the thesis is that it has proposed a novel approach for entropy estimation called the EC-kNN estimator which reduces the bias of the kNN estimator. The proposed EC-kNN comprises the local PCA learning and the boot-strap style correction acceptance procedure which together address the bias resulting from the uniformity assumption.

The EC-kNN has been shown to be useful in several simulation experiments from simple to complicated cases. The advantage of the EC-kNN has been shown to be prominent especially when the data set is high-dimensional and complicated. The experiments have implied that the local ellipsoidal correction and the boot-strap type acceptance procedure can properly capture the local uniformity region.

In this research, the EC-kNN estimator has been further applied to evaluation of MCMC posterior samples evaluation. It was shown to be a comparatively useful tool to not only monitor the convergence but also the correctness of the MCMC posterior samples. The proposed framework has been shown to be an appropriate tool to compare different samplers. Moreover, there are more applications other than evaluation of MCMC posterior samples requiring entropy estimation. The potential applications of this estimator can be extended in other research fields in the future.

One future direction of this research is exploring other potentials of modeling the local shape. This research utilizes an ellipsoid shape, which can be inflexible to catch the local structure in some complicated cases. Other shapes such as a convex hull can be potential candidates in further research.

Another potential route for the further research is the choice of number of neighbors k. An optimal choice of k should be capable of capturing the local non-uniformity, a smaller k can improve the efficiency but the learned local ellipsoid can be unrealistic, whereas a too big k can lead to poor efficiency and also an unrealistic local ellipsoid. In the simulation experiments of this research, the k is fixed to 25 in every dimension, and it turns out that the choice of k has worked well. However, a more sophisticated procedure of choosing of k can potentially improve the efficiency and the accuracy of the estimator.

Reference

Berger, A. L., Pietra, V. J. D., & Pietra, S. A. D. (1996). A maximum entropy approach to natural language processing. *Computational linguistics*, *22*(1), 39-71.

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, *112*(518), 859-877.

Calonico, S., Cattaneo, M. D., & Farrell, M. H. (2017). On the effect of bias estimation on coverage accuracy in nonparametric inference. *Journal of the American Statistical Association*, (just-accepted).

Chauveau, D., & Vandekerkhove, P. (2014). The Nearest Neighbor entropy estimate: an adequate tool for adaptive MCMC evaluation.

Colin Rundel, Mine Cetinkaya-Rundel, Merlise Clyde and David Banks (2017). statsr: Companion package for the Coursera Statistics with R specialization. R package version 0.0.1.

Cover, T. M., & Thomas, J. A. (2006). Elements of information theory 2nd edition.

Cowles, M. K., & Carlin, B. P. (1996). Markov chain Monte Carlo convergence diagnostics: a comparative review. *Journal of the American Statistical Association*, *91*(434), 883-904.

Cusumano-Towner, M. F., & Mansinghka, V. K. (2016). Quantifying the probable approximation error of probabilistic inference programs. *arXiv preprint arXiv:1606.00068*.

Dudewicz, E. J., & Van Der Meulen, E. C. (1981). Entropy-based tests of uniformity. *Journal of the American Statistical Association*, *76*(376), 967-974.

El Adlouni, S., Favre, A. C., & Bobée, B. (2006). Comparison of methodologies to assess the convergence of Markov chain Monte Carlo methods. *Computational Statistics & Data Analysis*, *50*(10), 2685-2701.

Gorham, J., & Mackey, L. (2015). Measuring sample quality with Stein's method. In *Advances in Neural Information Processing Systems* (pp. 226-234).

Fuhrman, S., Cunningham, M. J., Wen, X., Zweiger, G., Seilhamer, J. J., & Somogyi, R. (2000). The application of Shannon entropy in the identification of putative drug targets. *Biosystems*, *55*(1-3), 5-14.

Gulko, L. (1999). The entropy theory of stock option pricing. *International Journal of Theoretical and Applied Finance*, *2*(03), 331-355.

Gao, S., Ver Steeg, G., & Galstyan, A. (2015, February). Efficient estimation of mutual information for strongly dependent variables. In *Artificial Intelligence and Statistics* (pp. 277-286).

Hampe, J., Schreiber, S., & Krawczak, M. (2003). Entropy-based SNP selection for genetic association studies. *Human Genetics*, *114*(1), 36-43.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57*(1), 97-109.

Joe, H. (1989). Relative entropy measures of multivariate dependence. *Journal of the American Statistical Association*, *84*(405), 157-164.

Kraskov, A., Stögbauer, H., & Grassberger, P. (2004). Estimating mutual information. *Physical review E*, *69*(6), 066138.

kNN, N. N. (1987). Sample estimate of the entropy of a random vector. *Problemy Peredachi Informatsii*, *23*(2), 9-16.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The annals of mathematical statistics*, *22*(1), 79-86.

Itti, L., & Baldi, P. F. (2005). Bayesian surprise attracts human attention. In *Advances in neural information processing systems* (pp. 547-554).

Minka, T. P. (2001, August). Expectation propagation for approximate Bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*(pp. 362-369). Morgan Kaufmann Publishers Inc..

Murray, I., Prescott Adams, R., & MacKay, D. J. (2010). Elliptical slice sampling.

Neal, R. M. (2001). Annealed importance sampling. *Statistics and computing*, *11*(2), 125-139.

Noh, Y. K., Sugiyama, M., Liu, S., Plessis, M. C., Park, F. C., & Lee, D. D. (2014, April). Bias reduction and metric learning for nearest-neighbor estimation of Kullback-Leibler divergence. In *Artificial Intelligence and Statistics* (pp. 669-677).

Orava, Jan. "K-nearest neighbour kernel density estimation, the choice of optimal k." *Tatra Mountains Mathematical Publications* 50, no. 1 (2011): 39-50.

Pérez-Cruz, F. (2008, July). Kullback-Leibler divergence estimation of continuous distributions. In *Information Theory, 2008. ISIT 2008. IEEE International Symposium on* (pp. 1666-1670). IEEE.

Philippatos, G. C., & Wilson, C. J. (1972). Entropy, market risk, and the selection of efficient portfolios. *Applied Economics*, *4*(3), 209-220.

Raftery, A.E. and Lewis, S. (1992), How Many Iterations in the Gibbs Sampler? *Bayesian Statistics*, 4, 763-773.

Roberts, G.O. (1992), Convergence Diagnostics of the Gibbs Sampler. *Bayesian Statistics*, 4, 775-782.

Sasaki, H., Noh, Y. K., Niu, G., & Sugiyama, M. (2016). Direct density derivative estimation. *Neural computation*, *28*(6), 1101-1140.

Silverman, B. W. (1986). *Density estimation for statistics and data analysis*. CRC press.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell system technical journal*, *27*(3), 379-423.

Wang, Q., Kulkarni, S. R., & Verdú, S. (2009). Divergence estimation for multidimensional densities via $ k $-nearest-neighbor distances. *IEEE Transactions on Information Theory*, *55*(5), 2392-2405.

Lord, W. M., Sun, J., & Bollt, E. M. (2017). Geometric k-nearest neighbor estimation of entropy and mutual information. *arXiv preprint arXiv:1711.00748*.